UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/938,921	08/24/2001	Walter Clark Milliken	BBNT-P01-128	3501

28120 7590 07/30/2007

FISH & NEAVE IP GROUP
ROPES & GRAY LLP
ONE INTERNATIONAL PLACE
BOSTON, MA 02110-2624

EXAMINER	
NGUYEN, QUANG N	

ART UNIT	PAPER NUMBER
2141	

MAIL DATE	DELIVERY MODE
07/30/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/938,921
Filing Date: August 24, 2001
Appellant(s): MILLIKEN ET AL.

Edward J. Kelly
For Appellant

Examiner's Answer

This is in response to the appeal brief filed 04/02/2007 appealing from the Office action mailed 10/05/2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of the claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

The following is a listing of the prior art of record relied upon in the rejection of claims under appeal:

Curtis et al. (US 6,000,016) issued on 12/07/1999.

Zuraski, Jr. et al. (US 6,560,740) issued on 05/06/2003.

Nataraj et al. (US 6,757,779) issued on 06/29/2004.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 101

1. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any **new and useful** process, **machine**, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Claim 21 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

3. Claim 21 is directed to an arithmetic logic unit comprising a register unit; an operations unit; and a ternary content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit, hence, claim 21 falls under the category of a "machine". However, claim 21 is directed to non-statutory subject matter because it appears to have no substantial practical application. To satisfy section 35 USC § 101 requirements *(i.e., to satisfy the condition of a **new and useful machine**)*, the claim must be for a practical application of a 35 USC § 101 judicial exceptions. A practical application of a 35 USC § 101 judicial exception is claimed if the claimed invention physically transforms an article or physical object to a different state or thing, or if the claimed invention otherwise produces a useful, concrete, and tangible result.

Claim 21 does not appear to result in a physical transformation nor does it appear to produce a useful, concrete, and tangible result. Specially, it does not appear to produce a tangible result because merely describing *"a ternary content addressable memory coupled to a register unit and an operations unit within an arithmetic logic unit"*

fails to describe, to use, or to make available for use, the result of the description in order to enable its functionality and usefulness to be realized, i.e., **fails to describe a "new and useful machine"**. The practical application is not explicitly recited in the claim nor does it flow inherently therefrom, thus, claim 21 appears to be directed to non-statutory subject matter.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-6, 8-15 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Curtis et al. (US 6,000,016), hereinafter "Curtis", in view of Nataraj et al. (US 6,757,779), hereinafter "Nataraj".

6. As to claim 1, **Curtis** teaches a central processing unit CPU *(i.e., a microprocessor 80 as in Fig. 2)*, comprising:

an arithmetic logic unit *(ALU 0 and ALU 1 as illustrated in Fig. 2)*; and

a content addressable memory operatively coupled to the arithmetic logic unit within the CPU *(i.e., a bypass network 110 includes a content addressable memory*

array CAM 128 coupled to ALU 0 and ALU 1 within the microprocessor 80, as illustrated in Figs. 2-3 and 6) **(Curtis, col. 4, lines 31-48)** and configured to perform one or more matching operations *(wherein a piece of data can be stored from the output of an execution unit such as ALU 0, ALU1, to a register in bypass network 110 and read from the register in bypass network 110 to be make available to the same or another execution unit in one clock cycle, i.e., performing matching operations using address registers in bypass network 110)* **(Curtis, col. 3 lines 15-50 and col. 6, lines 1-31)**.

However, **Curtis** does not explicitly teach the content addressable memory is a ternary content addressable memory (ternary CAM or TCAM).

In an analogous art, **Nataraj** teaches a policy-based router can use a content addressable memory (CAM)-based system *(such as a ternary CAM system)* to implement a filtering or classification function to determine whether an incoming packet matches a policy statement *(i.e., matching network addresses)*, wherein the policy statements or policy words are stored in a ternary CAM array, which is one that is able to mask entries in a CAM array on a bit-by-bit basis *(for example, **Nataraj** teaches a ternary CAM array 304/404 is used to store policy statements and mask data to be compared with processed policy information of an incoming packet)* **(Nataraj, Figs. 3-4, col. 2, lines 12-32 and col. 7, lines 38-67)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the content addressable memory (CAM) is a ternary content addressable memory (TCAM), as disclosed by **Nataraj**, into the teaching of **Curtis**, because a ternary CAM is one that is

able to mask entries in a CAM array on a bit-by-bit basis *(also by definition, a single ternary entry "1XX" can represent 4 binary entries "100", "101", "110" and "111", hence, a ternary CAM would require a smaller number of table entries to represent each hierarchical address than a binary CAM)*, therefore, using a ternary CAM in place of a binary CAM would reduce a number of searches of the CAM needed in the worst case to find a matching entry **(Nataraj, col. 2, lines 12-32 and col. 7, lines 38-67)**.

7. As to claim 2, **Curtis-Nataraj** teaches the CPU of claim 1, wherein the one or more matching operations include a network packet processing operation ***(Curtis** teaches the CAM includes comparators that compare the destination address with a source address)* **(Curtis, Abstract, col. 2, lines 39-44, col. 6, lines 1-31 and Nataraj, col. 16, line 47 – col. 17, line 5)**.

8. As to claim 3, **Curtis-Nataraj** teaches the CPU of claim 2, wherein the packet processing operation includes an address lookup operation **(Curtis, Abstract, col. 2, lines 39-44, col. 6, lines 1-31 and Nataraj, col. 16, line 47 – col. 17, line 5)**.

9. As to claim 4, **Curtis-Nataraj** teaches the CPU of claim 3, wherein the address lookup operation includes an Internet Protocol (IP) address lookup operation **(Curtis, Abstract, col. 2, lines 39-44, col. 6, lines 1-31 and Nataraj, col. 16, line 47 – col. 17, line 5)**.

10. As to claim 5, **Curtis-Nataraj** teaches the CPU of claim 1, wherein the one or more matching operations include a packet stuff/unstuffy operation *(i.e., in order to perform matching operations, the data such as the policy statements/words or the source/destination addresses must be loaded/reloaded into address registers)* **(Curtis, Abstract, col. 2, lines 39-44, col. 6, lines 1-31 and Nataraj, col. 7, line 37 – col. 8, line 29, col. 9, lines 33-63 and col. 14, line 27 – col. 15, line 13)**.

11. As to claim 6, **Curtis-Nataraj** teaches the CPU of claim 1, wherein the one or more matching operations include a packet classification operation **(Curtis, Abstract, col. 2, lines 39-44, col. 6, lines 1-31 and Nataraj, col. 9, lines 33-63)**.

12. As to claim 8, **Curtis** teaches the CPU of claim 1, but does not further explicitly teach a first register configured to store a first 32-bit operand; and a second register configured to store a second 32-bit operand.

In an analogous art, **Nataraj** teaches a first register configured to store a first 32-bit operand *(i.e., comparand register C1 storing a first 32-bit comparand)*; and a second register configured to store a second 32 bit operand *(i.e., comparand register C2 storing a second 32-bit comparand, such that comparand register pair C1/C2 is coupled/configured to receive a 64 bit value from the data bus 1604 as in Fig. 21)* **(Nataraj, Fig. 21 and col. 37, line 46 – col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of a first register

configured to store a first 32-bit operand and a second register configured to store a second 32-bit operand, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

13. As to claim 9, **Curtis** teaches the CPU of claim 8, but does not explicitly teach the ternary content addressable memory performs the one or more matching operations based on at least one of the first or second 32-bit operands.

In an analogous art, **Nataraj** teaches the ternary content addressable memory performs the one or more matching operations based on at least one of the first or second 32-bit operands *(i.e., performing the one or more matching operations based on at least one of C1-C8 comparand registers)* **(Nataraj, Fig, 21 and col. 37, line 46 – col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory performs the one or more matching operations based on at least one of the first or second 32-bit operands, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each

different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

14. As to claim 10, **Curtis** teaches the CPU of claim 8, but does not explicitly teach the ternary content addressable memory includes a memory array including a group of 64-bit entries and wherein, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand.

In analogous art, **Nataraj** teaches the ternary content addressable memory includes a memory array including a group of 64-bit entries *(i.e., TCAM array 1501/1601 can be configured for x32, x64, x128, or x256 operation, so 64-bits entries will span 2 row segments, for example, S1/S2, S3/S4, S5/S6 and S7/S8 of row* $1522_{1\text{-}Y}$*)*, and wherein, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand *(i.e., when the system is configured in x64 mode, the 64-bit comparand word is loaded into all 4 comparand register segment pairs C1/C2, C3/C4, C5/C6, C7/C8 simultaneously for comparison with each of the 8 corresponding segments S1-S8 in each row of the TCAM array 1601 as in Fig. 21)* **(Nataraj, Figs. 15 and 21, col. 36, lines 45-62 and col. 37, line 62 - col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory includes a memory array including a group of 64-bit entries and wherein, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

15. As to claim 11, **Curtis** teaches the CPU of claim 1, but does not explicitly teach the ternary content addressable memory includes a memory array that includes a group of 64-bits entries.

In an analogous art, **Nataraj** teaches the ternary content addressable memory includes a memory array that includes a group of 64-bits entries *(i.e., TCAM array 1501/1601 can be configured for x32, x64, x128, or x256 operation, so 64-bits entries will span 2 row segments, for example, S1/S2, S3/S4, S5/S6 and S7/S8 of row* 1522_{1-Y}*)* **(Nataraj, Figs. 15 and 21, col. 22, lines 8-45 and col. 37, line 62 – col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory includes a memory array that includes a group of 64-bits entries, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

16. As to claim 12, **Curtis** teaches the CPU of claim 11, but does not explicitly teach the memory array comprises 32 entries.

In an analogous art, **Nataraj** teaches the memory array comprises 32 entries *(i.e., TCAM array 1501/1601 can be configured for x32, x64, x128, or x256 operation, so 64-bits entries will span 2 row segments of 32 CAM cells per row, for example, S1/S2, S3/S4, S5/S6 and S7/S8 of row* $1522_{1\text{-}Y}$*, hence, 32 entries of 64-bits will include 8 rows of the TCAM array 1501/1601)* **(Nataraj, Figs. 15 and 21, col. 22, lines 8-45 and col. 37, line 62 – col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory includes a memory array that includes 32 entries of 64-bits entries, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be

motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

17. As to claim 13, **Curtis** teaches the CPU of claim 1, but does not explicitly teach the ternary content addressable memory is configured to compare an operand to a group of entries.

In an analogous art, **Nataraj** teaches the ternary content addressable memory is configured to compare an operand to a group of entries *(i.e., the TCAM array 404 is configured to compare an operand 168.69.43.100 to a group of entries 168.0.0.0/8, 168.69.0.0/16, and 168.69.62.0/24 as illustrated in Fig. 11)* **(Nataraj, Fig. 11 and col. 16, line 47 – col. 17, line 5)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory is configured to compare an operand to a group of entries, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

18. As to claim 14, **Curtis** teaches the CPU of claim 13, but does not explicitly teach set a first flag when the operand fails to match an entry in the group of entries, and set a second flag when the operand matches multiple entries of the group of entries.

In an analogous art, **Nataraj** teaches the ternary content addressable memory is configured to set a first flag when the operand fails to match an entry in the group of entries, and set a second flag when the operand matches multiple entries of the group of entries *(i.e., the CAM device 1200 includes a TCAM array 1201, address logic 1209, comparand register 1207, etc., and further includes logic for generating match flag, multiple match flag and/or full-flag signals)* **(Nataraj, col. 17, lines 15-22 and col. 18, lines 14-28)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory is configured to set a first flag when the operand fails to match an entry in the group of entries, and set a second flag when the operand matches multiple entries of the group of entries, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

19. As to claim 15, **Curtis** teaches the CPU of claim 13, but does not explicitly teach the ternary content addressable memory is configured to sequentially load the group of entries from a succession of mask/value pairs transferred to the ternary content addressable memory.

In an analogous art, **Nataraj** teaches the ternary content addressable memory is configured to sequentially load the group of entries from a succession of mask/value pairs transferred to the ternary content addressable memory *(i.e., the TCAM array 404 is configured to sequentially load a group of entries such as 168.0.0.0/8, 168.69.0.0/16, and 168.69.62.0/24 into to compare with a comparand/search key 168.69.43.100 as illustrated in Fig. 11)* **(Nataraj, Fig. 11 and col. 16, line 47 – col. 17, line 5)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the ternary content addressable memory is configured to sequentially load the group of entries from a succession of mask/value pairs transferred to the ternary content addressable memory, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

20. As to claim 20, **Curtis-Nataraj** teaches a system for forwarding packets in a network device as in claim 1, comprising:

means for receiving at least one packet *(policy statement table 404 is stored in a ternary CAM array that stores policy statements/words)* **(Nataraj, col. 7, lines 38-67)**; and

means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device *(i.e., performing classifying and filtering policy statements using the ternary CAM 404)* **(Nataraj, col. 7, line 38 – col. 9, line 63)**.

21. Claims 7, 16 and 18-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Curtis, in view of Nataraj, and further in view of Zuraski, Jr. et al. (US 6,560,740), hereinafter "Zuraski".

22. As to claim 7, **Curtis-Nataraj** teaches the CPU of claim 1, but does not explicitly teach the ternary content addressable memory is located within the arithmetic logic unit.

In an analogous art, **Zuraski** teaches a content addressable memory CAM 82 is located within a repair logic unit 70, as illustrated in Fig. 8, to compare provided address signals with the contents of memory locations within the CAM 82 to determine a match **(Zuraski, Fig. 8 and col. 9, line 47 – col. 10, line 5)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of locating the

(ternary) content addressable memory within the (arithmetic) logic unit, as disclosed by **Zuraski**, into the teaching of **Curtis-Nataraj**. One would be motivated to do so *(i.e., to embed/integrate the ternary content addressable memory within the arithmetic logic unit)* to provide support hardware-based searching/matching engine functions by quickly examining incoming packets *(examining address information/signals)* and forwarding them to other systems in the network for further processing.

23. As to claim 16, **Curtis-Nataraj** teaches a method for processing packets in a network device such as the network device in claim 1, but does not explicitly teach the ternary content addressable memory resident within an arithmetic logic unit.

In an analogous art, **Zuraski** teaches a content addressable memory CAM 82 is located within a repair logic unit 70, as illustrated in Fig. 8, to compare provided address signals with the contents of memory locations within the CAM 82 to determine a match **(Zuraski, Fig. 8 and col. 9, line 47 – col. 10, line 5)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of locating the (ternary) content addressable memory within the (arithmetic) logic unit, as disclosed by **Zuraski**, into the teaching of **Curtis-Nataraj**. One would be motivated to do so *(i.e., to embed/integrate the ternary content addressable memory within the arithmetic logic unit)* to provide support hardware-based searching/matching engine functions by quickly examining incoming packets *(examining address information/signals)* and forwarding them to other systems in the network for further processing.

24. Claim 18 recites a method that contains substantially the same limitation as recited in claim 3; therefore, it is rejected under the same rationale.

25. Claim 19 recites a method that contains substantially the same limitation as recited in claim 6; therefore, it is rejected under the same rationale.

26. Claim 21 rejected under 35 U.S.C. 103(a) as being unpatentable over Zuraski, in view of Nataraj.

28. As to claim 21, **Zuraski** teaches an arithmetic logic unit *(such as a Built-In-Self-Test BIST logic unit 20 of Fig. 6)*, comprising:

a register unit *(a memory status register 68 of Fig. 6)*;

an operations unit *(a data comparator 66 of Fig. 6)*; and

a content addressable memory *(a repair LU 70 of Fig. 6 having CAM 82 as in Fig. 8)* coupled to the register unit and the operations unit within the arithmetic logic unit.

However, **Zuraski** does not explicitly teach the content addressable memory is a ternary content addressable memory (ternary CAM or TCAM).

In an analogous art, **Nataraj** teaches a policy-based router can use a content addressable memory (CAM)-based system *(such as a ternary CAM system)* to implement a filtering or classification function to determine whether an incoming packet matches a policy statement *(i.e., matching network addresses)*, wherein the policy statements or policy words are stored in a ternary CAM array, which is one that is able

to mask entries in a CAM array on a bit-by-bit basis *(for example, **Nataraj** teaches a ternary CAM array 304/404 is used to store policy statements and mask data to be compared with processed policy information of an incoming packet)* **(Nataraj, Figs. 3-4, col. 2, lines 12-32 and col. 7, lines 38-67)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the content addressable memory (CAM) is a ternary content addressable memory (TCAM), as disclosed by **Nataraj**, into the teaching of **Zuraski**, because **a ternary CAM is one that is able to mask entries in a CAM array on a bit-by-bit basis** *(also by definition, a single ternary entry "1XX" can represent 4 binary entries "100", "101", "110" and "111", hence, a ternary CAM would require a smaller number of table entries to represent each hierarchical address than a binary CAM)*, therefore, using a ternary CAM in place of a binary CAM would reduce a number of searches of the CAM needed in the worst case to find a matching entry **(Nataraj, col. 2, lines 12-32 and col. 7, lines 38-67)**.

(10) Response to Arguments

In the remarks, applicant argued in substance that

(**A**) *"Appellant notes that the Examiner does not address whether claim 21 falls under one of the four enumerated categories of patentable subject matter recited in 35 U.S.C. 101. Accordingly, the Examiner does not establish a prima facie basis for denying patentability under U.S.C. 101"*, as recited in page 4 of the Appeal Brief.

As to point (**A**), Examiner respectfully disagrees noting that claim 21 is directed to an arithmetic logic unit comprising a register unit; an operations unit; and a ternary content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit, hence, claim 21 falls under the category of a "machine". However, Examiner respectfully submits that claim 21 is directed to non-statutory subject matter because it appears to have no substantial practical application. To satisfy section 35 USC § 101 requirements *(i.e., to satisfy the condition of a **new and useful machine**)*, the claim must be for a practical application of a 35 USC § 101 judicial exceptions. A practical application of a 35 USC § 101 judicial exception is claimed if the claimed invention physically transforms an article or physical object to a different state or thing, or if the claimed invention otherwise produces a useful, concrete, and tangible result.

Claim 21 does not appear to result in a physical transformation nor does it appear to produce a useful, concrete, and tangible result. Specially, it does not appear to produce a tangible result because merely describing *"a ternary content addressable memory coupled to a register unit and an operations unit within an arithmetic logic unit"* fails to describe, to use, or to make available for use, the result of the description in order to enable its functionality and usefulness to be realized, i.e., **fails to describe a "new and useful machine"**. The practical application is not explicitly recited in the claim nor does it flow inherently therefrom, thus, claim 21 appears to be directed to non-statutory subject matter.

(**B**) Prior Arts do not disclose, or suggest *"a ternary content addressable memory operatively coupled to the arithmetic logic unit within the CPU and configured to perform one or more matching operations"*, as recited in page 6 of the Appeal Brief.

As to point (**B**), Examiner respectfully disagrees noting that **Curtis** teaches a content addressable memory operatively coupled to the arithmetic logic unit within the CPU *(i.e., a bypass network 110 includes a content addressable memory array CAM 128 coupled to ALU 0 and ALU 1 within the microprocessor 80, as illustrated in Figs. 2-3 and 6)* **(Curtis, col. 4, lines 31-48)** and configured to perform one or more matching operations *(wherein a piece of data can be stored from the output of an execution unit such as ALU 0, ALU1, to a register in bypass network 110 and read from the register in bypass network 110 to be make available to the same or another execution unit in one clock cycle, i.e., performing matching operations using address registers in bypass network 110)* **(Curtis, col. 3 lines 15-50 and col. 6, lines 1-31)**.

However, **Curtis** does not explicitly teach the content addressable memory is a ternary content addressable memory (ternary CAM or TCAM).

In an analogous art, **Nataraj** teaches a policy-based router can use a content addressable memory (CAM)-based system *(such as a ternary CAM system)* to implement a filtering or classification function to determine whether an incoming packet matches a policy statement *(i.e., matching network addresses)*, wherein the policy statements or policy words are stored in a ternary CAM array, which is one that is able to mask entries in a CAM array on a bit-by-bit basis *(for example, **Nataraj** teaches a ternary CAM array 304/404 is used to store policy statements and mask data to be*

compared with processed policy information of an incoming packet) **(Nataraj, Figs. 3-4, col. 2, lines 12-32 and col. 7, lines 38-67)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the content addressable memory (CAM) is a ternary content addressable memory (TCAM), as disclosed by **Nataraj**, into the teaching of **Curtis**, because **a ternary CAM is one that is able to mask entries in a CAM array on a bit-by-bit basis** *(also by definition, a single ternary entry "1XX" can represent 4 binary entries "100", "101", "110" and "111", hence, a ternary CAM would require a smaller number of table entries to represent each hierarchical address than a binary CAM)*, therefore, using a ternary CAM in place of a binary CAM would reduce a number of searches of the CAM needed in the worst case to find a matching entry **(Nataraj, col. 2, lines 12-32 and col. 7, lines 38-67)**.

(**C**) *"Appellant submits that the Examiner's motivation is merely a conclusory statement regarding an allegedly benefit of incorporating a ternary content addressable memory into Curtis et al. Such motivation statements are insufficient for establishing a prima facie case of obviousness"*, as recited in page 7 of the Appeal Brief.

As to point (**C**), in response to applicant's argument that there is no suggestion to combine the references, Examiner respectfully submits that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one

of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992).

In this case, **Nataraj** teaches a policy-based router can use a content addressable memory (CAM)-based system *(such as a ternary CAM system)* to implement a filtering or classification function to determine whether an incoming packet matches a policy statement *(i.e., matching network addresses)*, wherein the policy statements or policy words are stored in a ternary CAM array, **which is one that is able to mask entries in a CAM array on a bit-by-bit basis** *(for example, **Nataraj** teaches a ternary CAM array 304/404 is used to store policy statements and mask data to be compared with processed policy information of an incoming packet)* **(Nataraj, Figs. 3-4, col. 2, lines 12-32 and col. 7, lines 38-67)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the content addressable memory (CAM) is a ternary content addressable memory (TCAM), as disclosed by **Nataraj**, into the teaching of **Curtis**, because **a ternary CAM is one that is capable of locally masking each entry in a CAM array on a bit-by-bit basis** *(also by definition, a single ternary entry "1XX" can represent 4 binary entries "100", "101", "110" and "111", hence, a ternary CAM would require a smaller number of table entries to represent each hierarchical address than a binary CAM)*, therefore, using a ternary CAM in place of a binary CAM would reduce a number of searches of the CAM needed in the worst case to find a matching entry **(Nataraj, col. 2, lines 12-32 and col. 7, lines 38-67)**.

(**D**) *"Appellant submits that the Examiner's purported motivation to combine the cited references is merely conclusory and based on impermissible hindsight"*, as recited in page 8 of the Appeal Brief.

As to point (**D**), in response to applicant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, Examiner respectfully submits that it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

(**E**) Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do not disclose or suggest, *"the one or more operations includes a packet stuff/unstuff operation"* of claim 5, as recited in page 8 of the Appeal Brief.

As to point (**E**), Examiner respectfully disagrees noting that in this case, **Curtis** teaches the CAM includes comparators that compare the destination address with a source address *(hence, performing matching operations by loading/unloading, i.e., stuffing/unstuffing, the source and destination addresses into the address registers)* **(Curtis, Abstract and col. 6, lines 1-31)**.

In addition, **Nataraj** teaches a classification or filtering system 400 for a policy-based router that use a ternary CAM array to store policy statements or policy words

(Nataraj, col. 7, line 37 – col. 8, line 29), wherein the policy field information is provided to the system 400 as policy search key 409 and the policy fields of policy search key 409 are compared with the policy statements stored in ternary CAM array 404 *(hence, performing matching operations by loading/unloading, i.e., stuffing/unstuffing, the policy statements/words into registers of ternary CAM array)* **(Nataraj, col. 9, lines 33-63)**.

Thus, Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do disclose or suggest, *"the one or more operations includes a packet stuff/unstuffy operation"*, as claimed in claim 5 of the invention.

(**F**) Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do not disclose or suggest, *"the one or more operations includes a packet classification operation"* of claim 6, as recited in page 11 of the Appeal Brief

As to point (**F**), Examiner respectfully disagrees noting that in this case, **Curtis** teaches the CAM includes comparators that compare the destination address with a source address *(hence, performing matching operations by comparing the source and destination addresses, i.e., performing packet classification operation)* **(Curtis, Abstract and col. 6, lines 1-31)**.

In addition, **Nataraj** teaches a classification or filtering system 400 for a policy-based router that use a ternary CAM array to store policy statements or policy words **(Nataraj, col. 7, line 37 – col. 8, line 29)**, wherein the policy field information is provided to the system 400 as policy search key 409 and the policy fields of policy

search key 409 are compared with the policy statements stored in ternary CAM array 404 *(hence, performing matching operations by comparing the policy statements/words, i.e., performing packet classification operation)* **(Nataraj, col. 9, lines 33-63)**.

Thus, Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do disclose or suggest, *"the one or more operations includes a packet classification operation"*, as claimed in claim 6 of the invention.

(**G**) Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do not disclose or suggest, *"a first register configured to store a first 32-bit operand; and a second register configured to store a second 32-bit operand"* of claim 8, as recited in page 15 of the Appeal Brief.

As to point (**G**), Examiner respectfully disagrees noting that **Curtis** teaches the CPU of claim 1 with the CAM including a plurality of registers **(Curtis, Figs. 3 and 6, col. 6, lines 1-31)**, but does not further explicitly teach a first register configured to store a first 32-bit operand; and a second register configured to store a second 32-bit operand.

However, in an analogous art, **Nataraj** teaches a first register configured to store a first 32-bit operand *(i.e., comparand register C1 storing a first 32-bit comparand)*; and a second register configured to store a second 32 bit operand *(i.e., comparand register C2 storing a second 32-bit comparand, such that comparand register pair C1/C2 is coupled/configured to receive a 64 bit value from the data bus 1604 as in Fig. 21)* **(Nataraj, Fig. 21 and col. 37, line 46 – col. 38, line 24)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of a first register configured to store a first 32-bit operand and a second register configured to store a second 32-bit operand, as disclosed by **Nataraj**, into the teaching of **Curtis**. One would be motivated to do so to allow the system to configure and enable a single CAM array to store and maintain a different table size in each different mode of operation, hence, the processing overhead in searching/matching for any word in excess of 32-bits to be dramatically improved by the flexibility configuration of the CAM array **(Nataraj, col. 22, lines 8-45)**.

Hence, Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do disclose or suggest, *"a first register configured to store a first 32-bit operand; and a second register configured to store a second 32-bit operand"*, as claimed in claim 8 of the invention.

(H) Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do not disclose or suggest, *"means for receiving at least one packet; and means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device"* of claim 20, as recited in page 18 of the Appeal Brief.

As to point (**H**), Examiner respectfully disagrees noting that, **Curtis** in view of **Nataraj** teaches a system for forwarding packets in a network device as in claim 1, further comprising:

means for receiving at least one packet *(policy statement table 404 is stored in a ternary CAM array that stores policy statements/words)* **(Nataraj, col. 7, lines 38-67)**; and

means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device *(i.e., performing classifying and filtering policy statements using the ternary CAM 404)* **(Nataraj, col. 7, line 38 – col. 9, line 63)**.

Hence, Prior Arts (**Curtis** and **Nataraj**, whether taken alone or in any reasonable combination) do disclose or suggest, *"means for receiving at least one packet; and means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device"* as claimed in claim 20.

(I) Prior Arts (**Curtis**, **Nataraj** and **Zuraski**, whether taken alone or in any reasonable combination) do not disclose or suggest, *"the ternary content addressable memory is located within the arithmetic logic unit"* of claim 7 and claim 16, as recited in page 21 and page 23 of the Appeal Brief.

As to point (I), Examiner respectfully disagrees noting that **Curtis-Nataraj** teaches the CPU of claim 1, but does not explicitly teach the ternary content addressable memory is located within the arithmetic logic unit.

However, in an analogous art, **Zuraski** teaches **a content addressable memory CAM 82 is located within a repair logic unit 70**, as illustrated in Fig. 8, to compare

provided address signals with the contents of memory locations within the CAM 82 to determine a match **(Zuraski, Fig. 8 and col. 9, line 47 – col. 10, line 5)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of locating the (ternary) content addressable memory within the (arithmetic) logic unit, as disclosed by **Zuraski**, into the teaching of **Curtis-Nataraj**. One of ordinary skill in the art would be motivated to do so *(i.e., to embed/integrate the ternary content addressable memory within the arithmetic logic unit)* to provide support hardware-based searching/matching engine functions by quickly examining incoming packets *(examining address information/signals)* and forwarding them to other systems in the network for further processing.

In response to the Appellant's additional argument that *"**Zuraski**, Jr. et al. does not disclose or suggest that repair logic unit 70 is an arithmetic logic unit"* (as recited in page 22 of the Appeal Brief), Examiner respectfully submits that one of ordinary skill in the art at the time of the invention would have duly recognized that **Zuraski's** disclosed mechanism for comparing provided address signals with the contents of memory locations would require the use of a content addressable memory CAM 82 located within a repair logic unit 70 in order to perform the matching operations as intended **(Zuraski, Fig. 8 and col. 9, line 47 – col. 10, line 5)**. In absence of any disclosure by the Appellant of specifically why using an arithmetic logic unit provides any sort of an advantage, is used for a particular purpose, or solves a specific problem, consequently, one of ordinary skill in the art would have readily found that it is obvious to incorporate

the feature of locating/embedding the content addressable memory within the logic unit, as disclosed by **Zuraski**, into the teachings of **Curtis-Nataraj** to achieve the claimed invention as recited in claim 7 and claim 16.

(**J**) Prior Arts (**Zuraski** and **Nataraj**, whether taken alone or in any reasonable combination) do not disclose or suggest the features of claim 21, as recited in page 26 of the Appeal Brief.

As to point (**J**), Examiner respectfully disagrees noting that **Zuraski** teaches an arithmetic logic unit *(such as a Built-In-Self-Test BIST logic unit 20 of Fig. 6)*, comprising:

a register unit *(a memory status register 68 of Fig. 6)*;

an operations unit *(a data comparator 66 of Fig. 6)*; and

a content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit *(a repair LU 70 having embedded CAM 82, as illustrated in Fig. 8, coupled with a memory status register 68 and a data comparator 66)* **(Zuraski, Figs. 6 and 8; col. 9, line 47 – col. 10, line 5)**.

However, **Zuraski** does not explicitly teach the content addressable memory is a ternary content addressable memory (ternary CAM or TCAM).

In an analogous art, **Nataraj** teaches a policy-based router can use a content addressable memory (CAM)-based system *(such as a ternary CAM system)* to implement a filtering or classification function to determine whether an incoming packet matches a policy statement *(i.e., matching network addresses)*, wherein the policy

statements or policy words are stored in a ternary CAM array, which is one that is able to mask entries in a CAM array on a bit-by-bit basis *(for example, **Nataraj** teaches a ternary CAM array 304/404 is used to store policy statements and mask data to be compared with processed policy information of an incoming packet)* **(Nataraj, Figs. 3-4, col. 2, lines 12-32 and col. 7, lines 38-67)**.

Therefore, it would have been obvious to one of ordinary skill in the Data Processing art at the time of the invention to incorporate the feature of the content addressable memory (CAM) is a ternary content addressable memory (TCAM), as disclosed by **Nataraj**, into the teaching of **Zuraski**, because **a ternary CAM is one that is able to mask entries in a CAM array on a bit-by-bit basis** *(also by definition, a single ternary entry "1XX" can represent 4 binary entries "100", "101", "110" and "111", hence, a ternary CAM would require a smaller number of table entries to represent each hierarchical address than a binary CAM)*, therefore, using a ternary CAM in place of a binary CAM would reduce a number of searches of the CAM needed in the worst case to find a matching entry **(Nataraj, col. 2, lines 12-32 and col. 7, lines 38-67)**.

For the above reasons, it is believed that the rejections should be sustained.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

Respectfully submitted,

Quang N. Nguyen

Conferees,

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

FISH & NEAVE IP GROUP, ROPES & GRAY LLP
One International Place
Boston, Massachusetts 02110-2624
(617) 951-7000 (Telephone)
(617) 951-7050 (Fax)